

FIG. 1
PRIOR ART

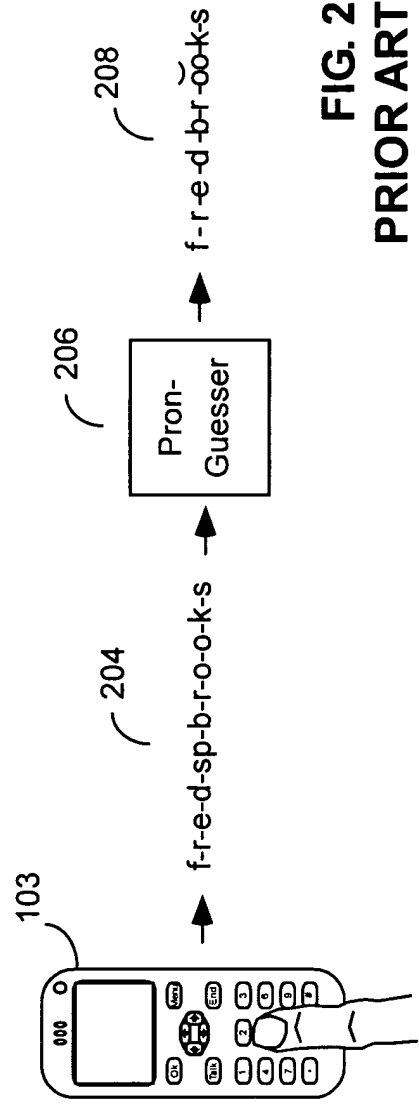


FIG. 2
PRIOR ART

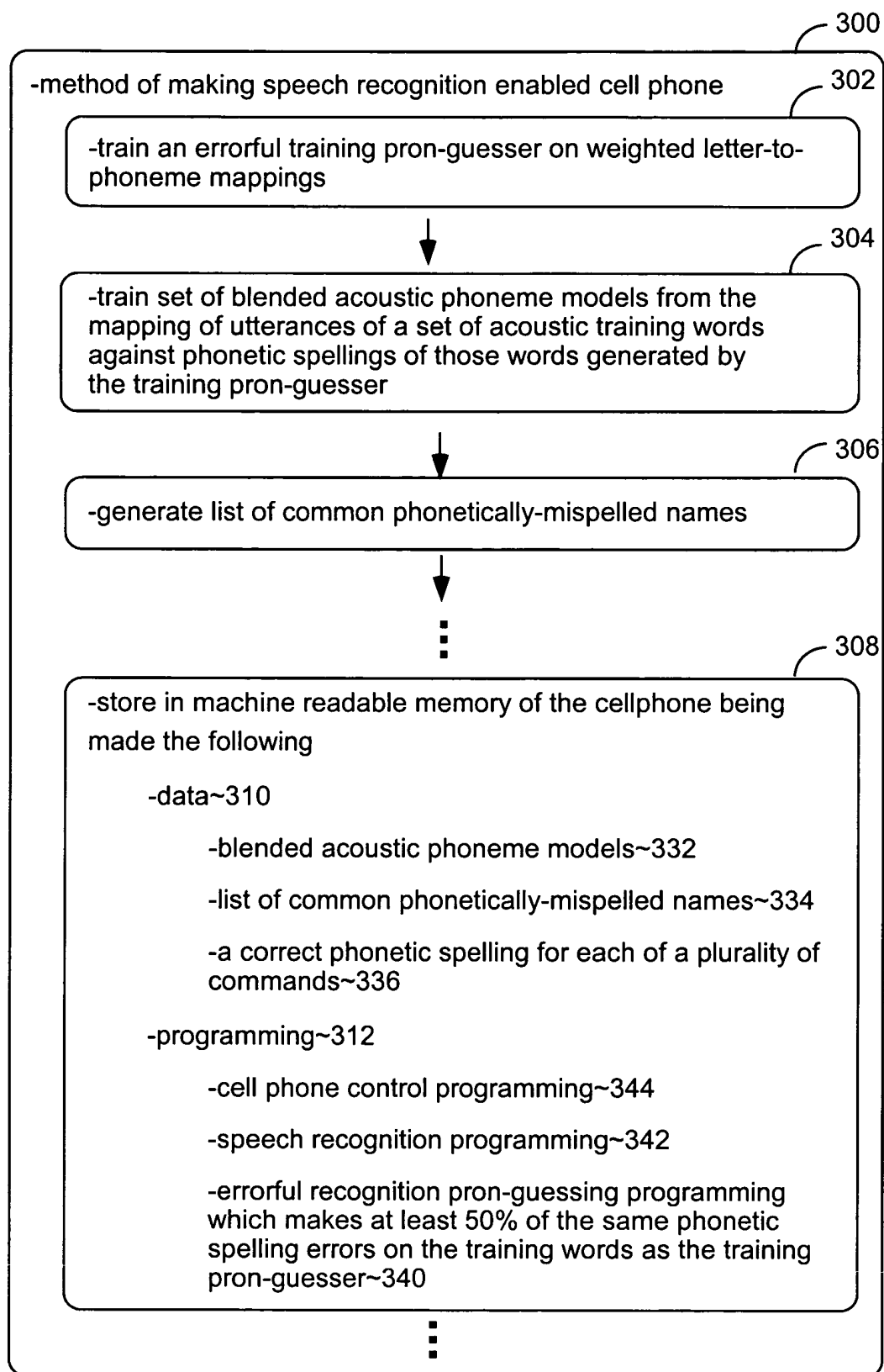


FIG. 3

- Training pron-guesser~302
 - for each of a set of pron-guesser training words, obtaining ~402
 - a textual spelling for the word~404
 - a phonetic spelling for the word~406
 - frequency with which the word occurs~408
 - for each training word~410
 - map letters of word's textual spelling against phonemes of the phonetic spelling for the word~412
 - for each letter in the word~414
 - create an associated letter context that contains~416
 - the letter~418
 - the phoneme it is mapped to in the word~420
 - the context of letters, if any, that occur near the letter in the word~422
 - a frequency weight calculated as a function of the frequency of the word raised to a power less than one~424
 - bin together all occurrences of identical letter contexts from all the training words~426
 - for each set of identical letter contexts~428
 - create a summed frequency weight equal to the sum of all the frequency weights from all letter context in set~430
 - for each letter~432
 - call a pronGuesserDTreeTraining for the set of letter contexts created for the letter, with its summed frequency weight~434

FIG. 4

FREQ.	WEIGHT	TEXT SPELLING	PHONETIC SPELLING
683	96	Josette	jh ow s eh t
...
320	57	Swanton	s w aa n t ax n
...
155	34	Fontenette	f aa n t ax n eh t
...
33	12	Parsonage	p aa r s ax n ax jh
...

FIG. 5

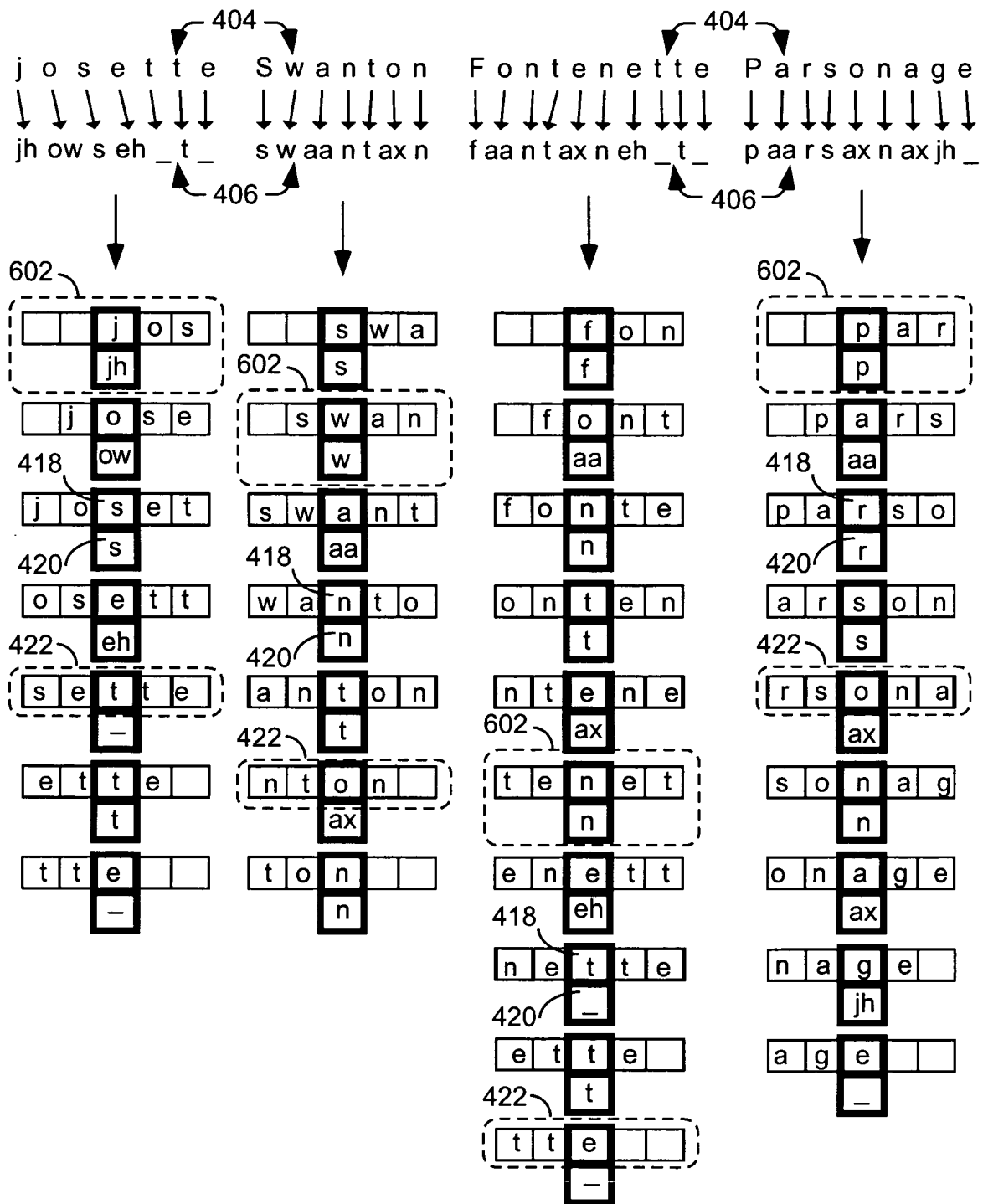


FIG. 6

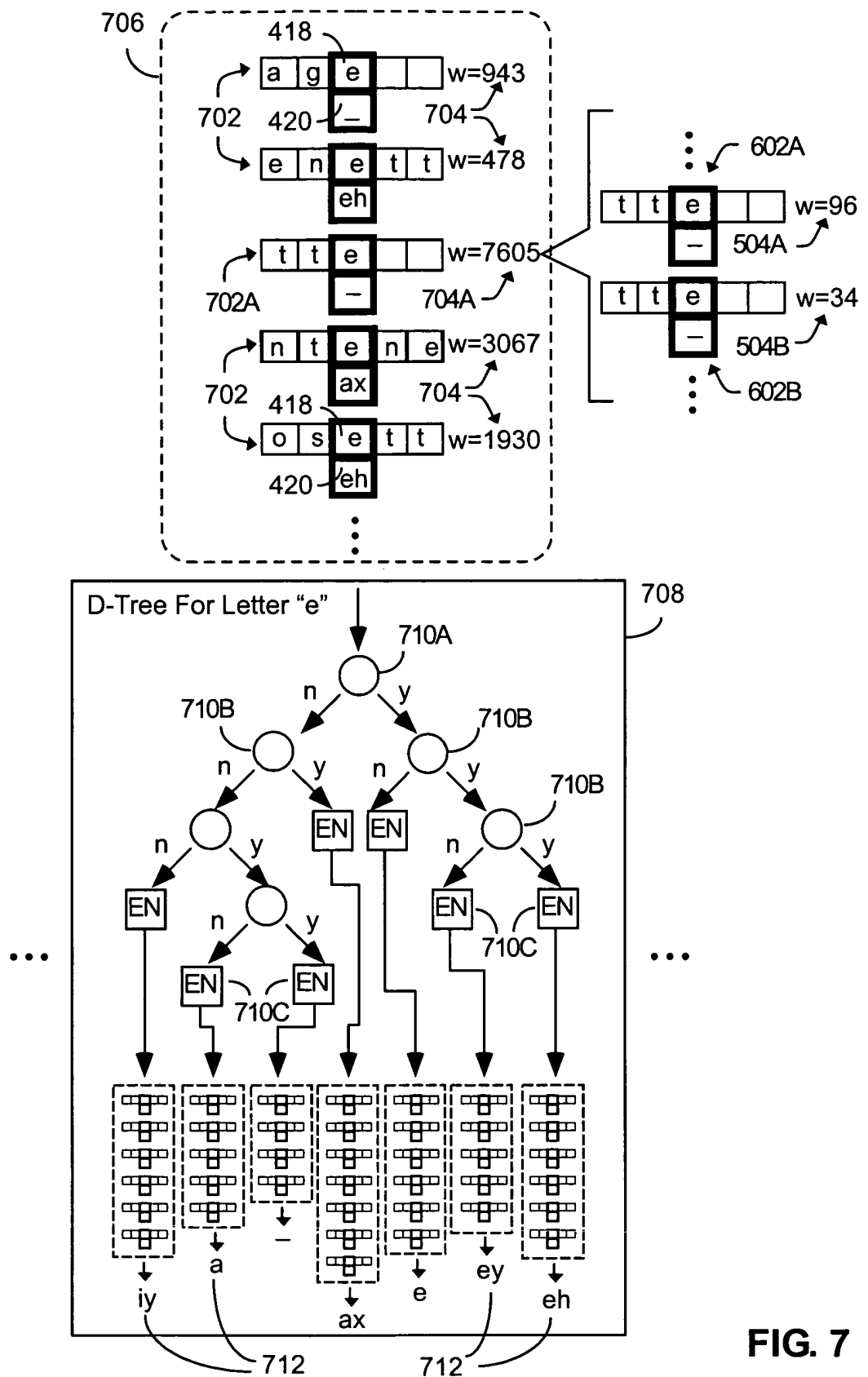


FIG. 7

- pronGuesserDTreeTraining~800
 - call processD-TreeNode with~802
 - currentNode equal to the d-tree's root node~804
 - currentContextSet equal to the set of contexts supplied to the d-tree's root~806

FIG. 8

- processDTreeNode (currentContextSet, currentNode)~900
 - for each possible context question~902
 - divide the current context set into two new context sets, one of contexts that satisfy the question and one of contexts that do not~904
 - for each of the two new context sets~906
 - calculate the entropy of the context set based on the distribution phonemes associated with the contexts of the set, weighting the contribution of each context's phoneme to the distribution as a function of the context's associated frequency based weight~900
 - if this total entropy for the two new context sets is the lowest calculated so far for the current node~908
 - save the entropy value as minEntropy~910
 - save the current context question as bestQuestion~912
 - save the two new context sets~914
 - if the difference between the entropy of the currentContextSet and minEntropy is not greater than a given amount~916
 - make the new node a terminal node~918
 - set the outputPhoneme for the terminal node equal to the phoneme associated with the largest number contexts in the node's context set~920
 - else~921
 - created a new node for each of the two new context sets~922
 - for each of two new nodes~924
 - if the number of contexts in the new node's context set is above a given number~926
 - call processDTreeNode with~928
 - currentNode equal to new node~930
 - currentContextSet equal to the new node's context set~932
 - else~934
 - make the new node a terminal node~936
 - set the outputPhoneme for the terminal node equal to the phoneme associated with the largest number contexts in the node's context set~938

FIG. 9

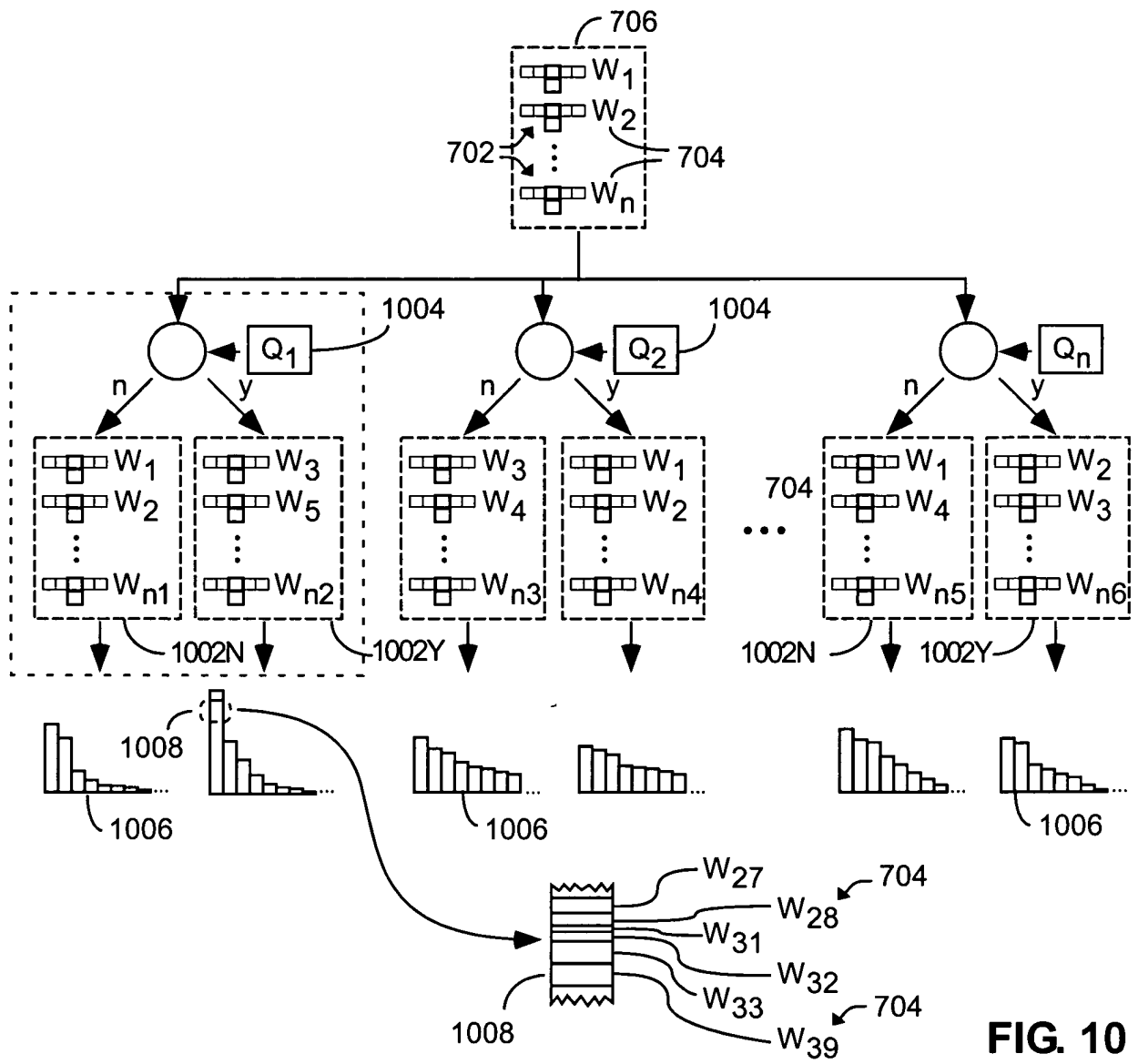


FIG. 10

- pron-guessing programming~1100
 - for each successive letter in the spelling of the word for which phonetic spelling is to be produced~1106
 - create a mapping context of the type used in training the pron-guesser's d-trees for the letter~1108
 - send the letter context through the d-tree corresponding to the given letter~1110
 - add the output phoneme associated with the end node of the d-tree into which the letter context is mapped to the phonetic spelling being created for the name~1112

FIG. 11

Correct pronunciations:

Cheryl	sh eh r ax l
Igor	iy g ao r
Jose	hh ow z ey
Philippe	f ih l iy p
Rene	r ax n ey

Incorrect guessed pronunciations:

Cheryl	ch eh r ax l	(starts with "ch" sound instead of "sh")
Igor	iy g ax r	(pronounced as "eager")
Jose	jh ow z ey	(starts with "j" sound instead of "h")
Philippe	f ih l ih p	(pronounced as "Philip")
Rene	r ax n	(final "e" is dropped)

FIG. 12

aa	sound of "o" in "pot"
ae	sound of "a" in "at"
ah	sound of "u" in "but"
ax	sound of "a" in "about" (schwa)
ay	sound of "i" in "night"
b	sound of "b" in "boy"
ch	sound of "ch" in "children"
d	sound of "d" in "day"
eh	sound of "e" in "bet"
ey	sound of "a" in "fake"
f	sound of "f" in "food"
hh	sound of "h" in "hey"
ih	sound of "i" in "bit"
iy	sound of "ee" in "bee"
jh	sound of "j" in "judge"
l	sound of "l" in "light"
m	sound of "m" in "may"
n	sound of "n" in "new"
ow	sound of "o" in "no"
p	sound of "p" in "pay"
r	sound of "r" in "ray"
s	sound of "s" in "so"
sh	sound of "sh" in "shame"
t	sound of "t" in "time"
uh	sound of "oo" in "hood"
v	sound of "v" in "very"
w	sound of "w" in "will"
y	sound of "y" in "yes"
z	sound of "z" in "zoo"
zh	sound of "si" in "Asia"

FIG. 13

error %	% of errors	Pho- neme																	
0.0	0.0	d	->	d	Ins	jh													
0.0	0.0	m	->	m	Ins														
0.0	0.0	p	->	p															
0.0	0.0	f	->	f	p														
0.1	0.0	v	->	v	Ins														
0.1	0.1	b	->	b	Ins	w													
0.1	0.3	r	->	r	Ins	t													
0.1	0.3	l	->	l	y	Ins	aa	ah	ao	ax									
0.3	0.5	t	->	t	th	Ins	ch	ey	ax	ay	ow								
0.3	0.7	n	->	n	Ins	ng													

3.5	5.1	s	->	s	z	Ins	sn	o	ey	iy	ow	ay								
4.2	0.4	uh	->	uh	ax	uw	ah													
4.7	1.0	jh	->	jh	g	hh	y	zh	Ins	ao	ax									
5.6	3.0	z	->	z	s	Ins														
5.8	5.3	aa	->	aa	ax	ow	ae	ao	ey	ah	Ins	n	uh							
5.9	1.0	ch	->	ch	sh	k	hh	t	Ins	s										
6.3	6.3	eh	->	eh	ax	iy	Ins	ey	ae	ih	aa	ay	sh							
6.7	1.1	y	->	y	Ins	iy	l	ay	g	ax	ey	zh								
7.3	22.1	ax	->	ax	ah	aa	eh	ae	ih	ey	Ins	iy	ao	ow	uh	uw	ay	f	jh	n
7.6	3.1	ey	->	ey	ax	ae	iy	Ins	ay	eh	aa	uw	ih							
8.8	2.9	ay	->	ay	iy	ih	ey	ax	y	zh	eh									
10.8	9.2	ae	->	ae	ax	aa	ey	ao	eh	Ins	ih									
14.7	0.1	zh	->	zh	iy	s														
29.8	14.9	ah	->	ah	ax	uw	Ins	ow	aa	ao	eh	ey	uh	y						

1404

1402

1406

1410A

1410

1412

FIG. 14

[illegible]

FIG. 15

- training blended acoustic phoneme models~1600
 - for each of a plurality of acoustic training words~dd02
 - have errorful pron-guessing programming generate a phonetic spelling for word~dd04
 - create an untrained associate word model for the word, composed of a sequence of untrained acoustic phoneme models corresponding to the word's phonetic spelling~dd06
 - repeatedly~dd08
 - use HMM to map one or more utterances of each training word against the acoustic model for that word~dd10
 - alter each acoustic phoneme model to be more like the utterance sounds mapped against it in all training words in which it occurs~dd12
 - for each phoneme~dd14
 - for each node number $n = 1$ through 3~dd16
 - call acoustic node clustering d-tree algorithm for the n th node of the set of all triphone model for the given phoneme~dd18

FIG. 16

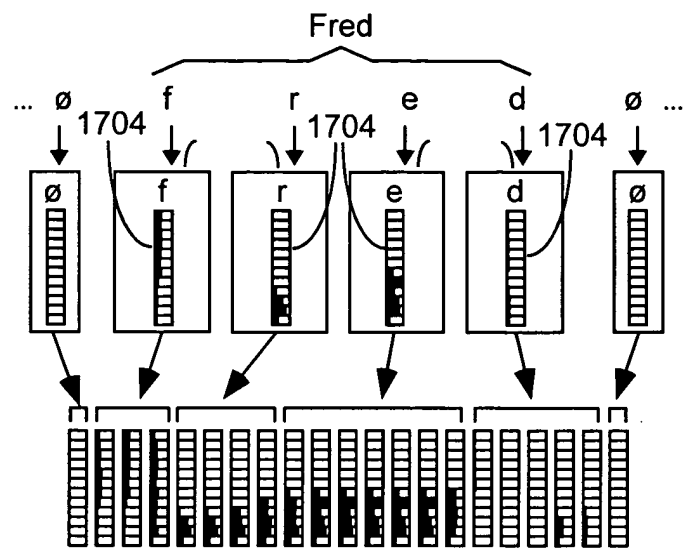


FIG. 17

1704

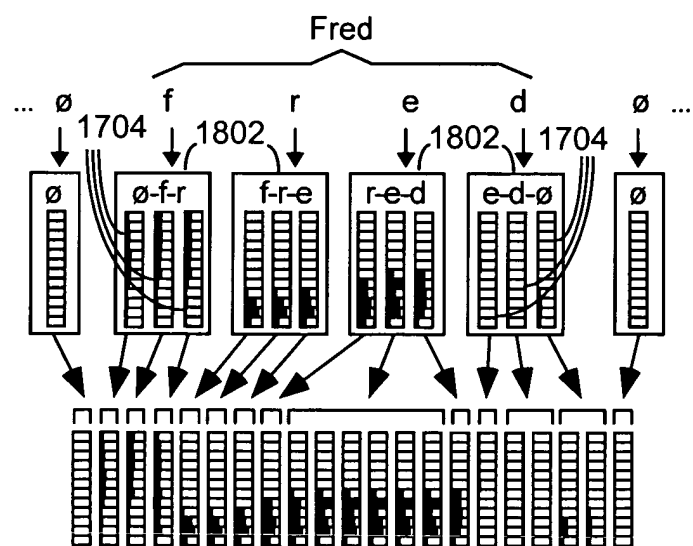


FIG. 18

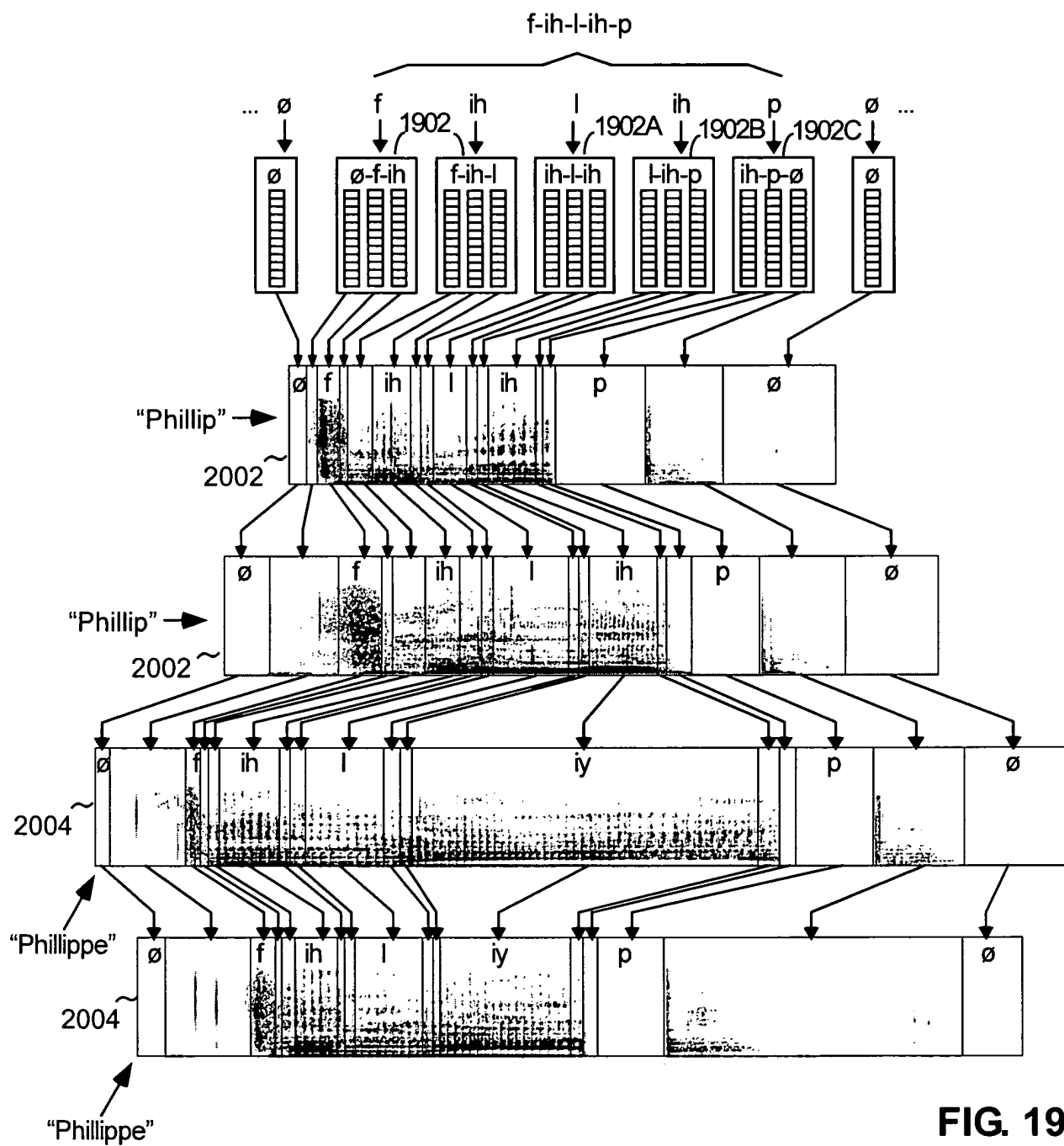


FIG. 19

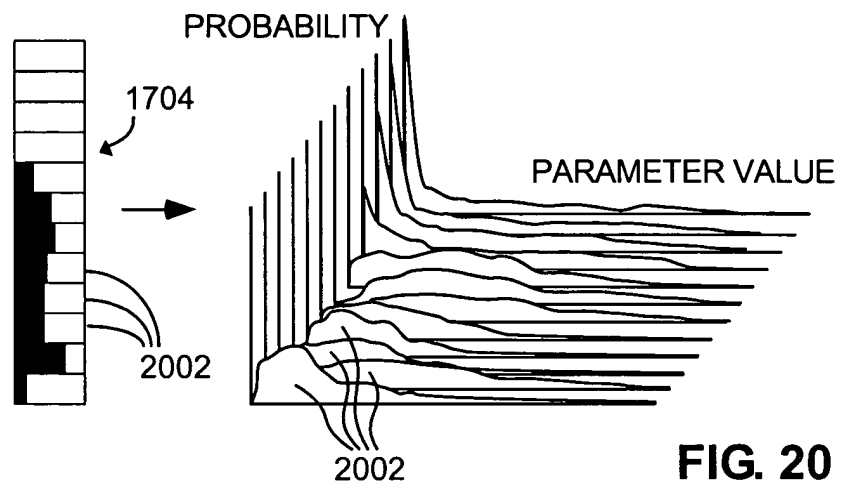


FIG. 20

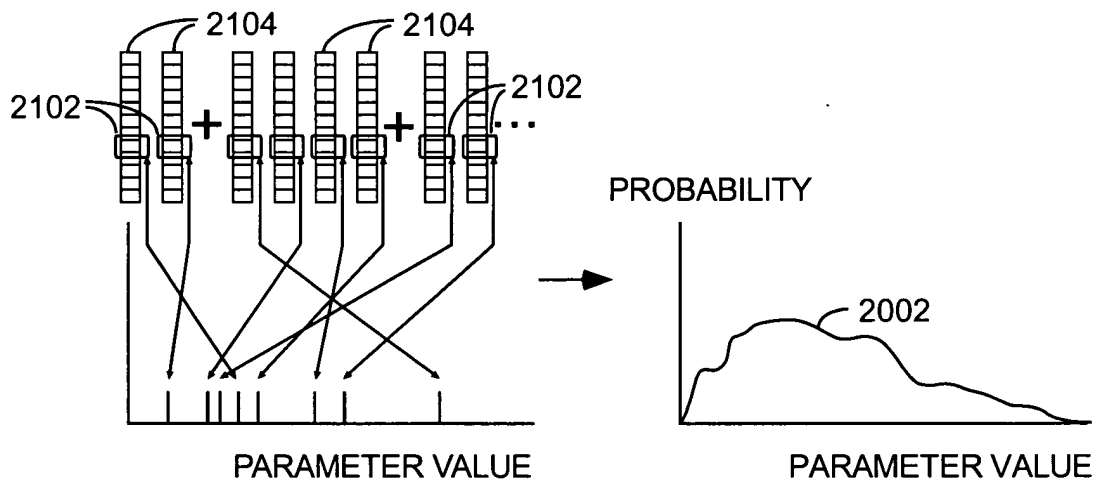


FIG. 21

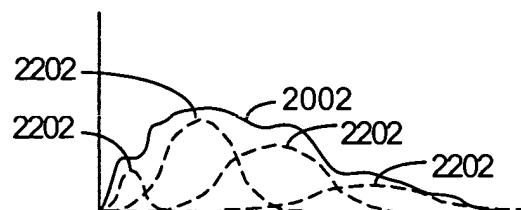


FIG. 22

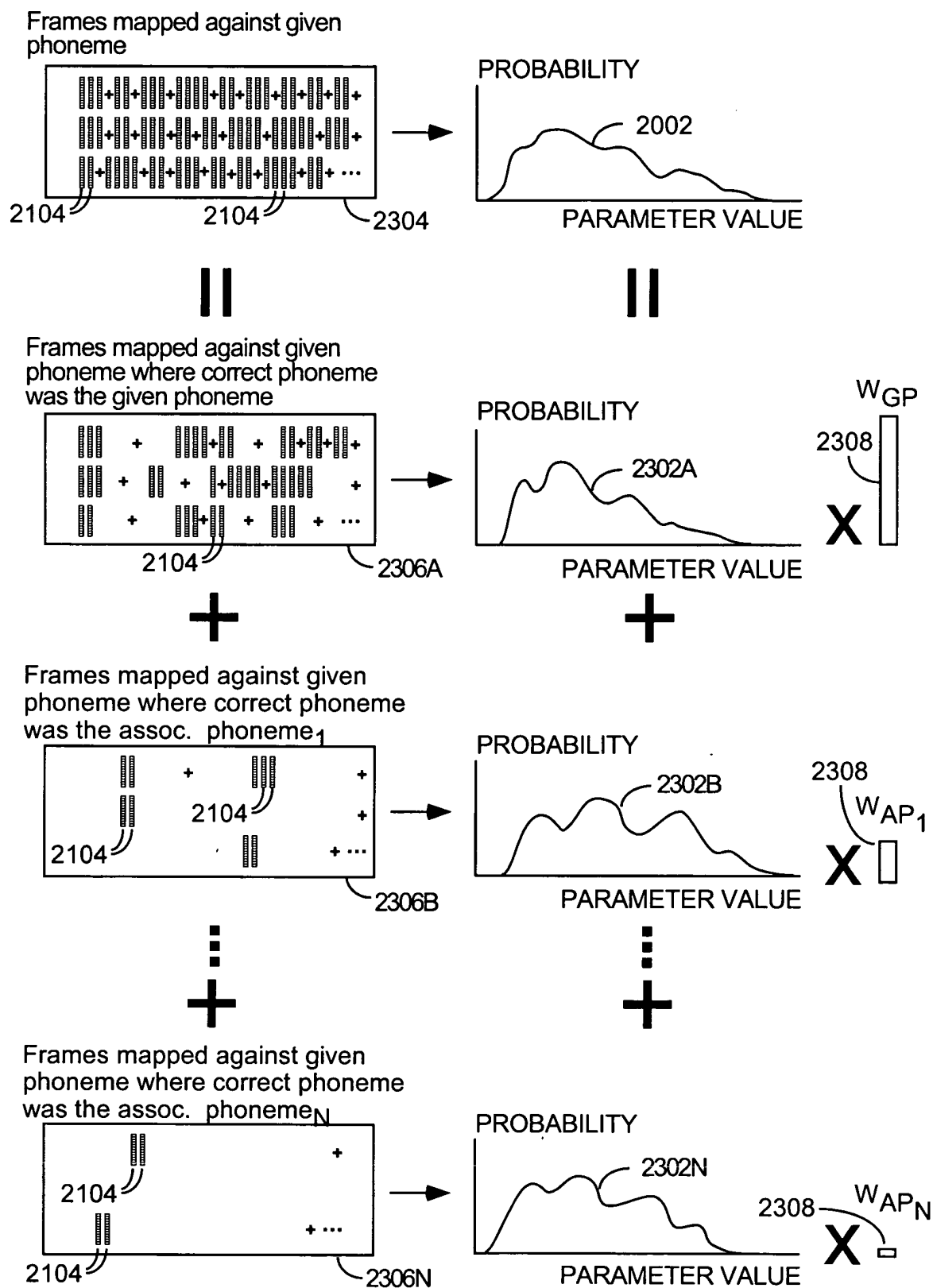


FIG. 23

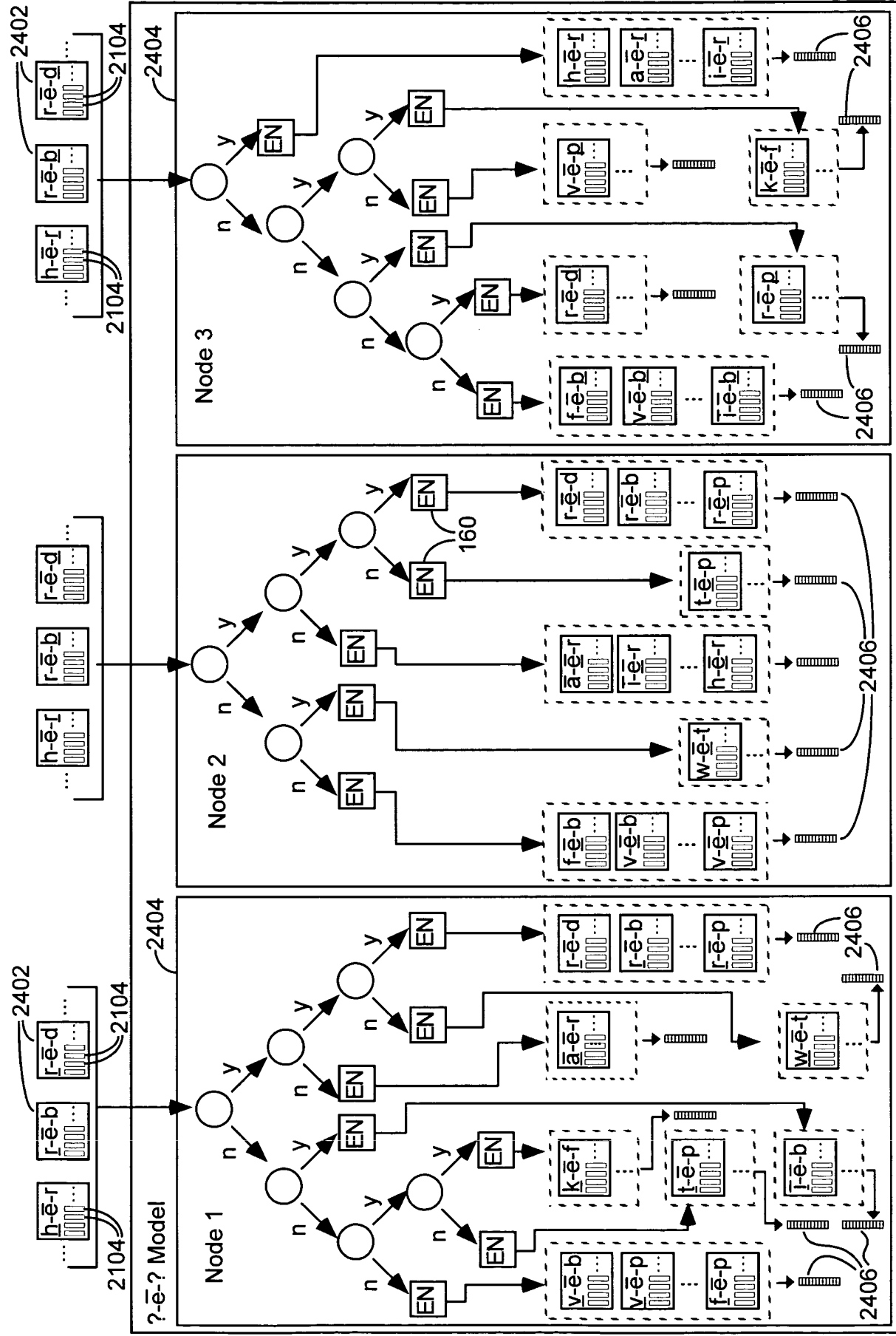


FIG. 24

- creating list of common phonetically-misspelled names~306
 - for each of a set of common names~2502
 - compare guessed phonetic spelling produced by recognition pron-guessers with known phonetic spellings for name~2504
 - if the guessed phonetic spelling is wrong, mark word as phonetically misspelled~2506
 - for each of a given number of the most common of the names marked as phonetically misspelled~2508
 - add to the list of common phonetically misspelled names~2510
 - the spelling of the name~2512
 - the correct phonetic spelling of the name~2514

FIG. 25

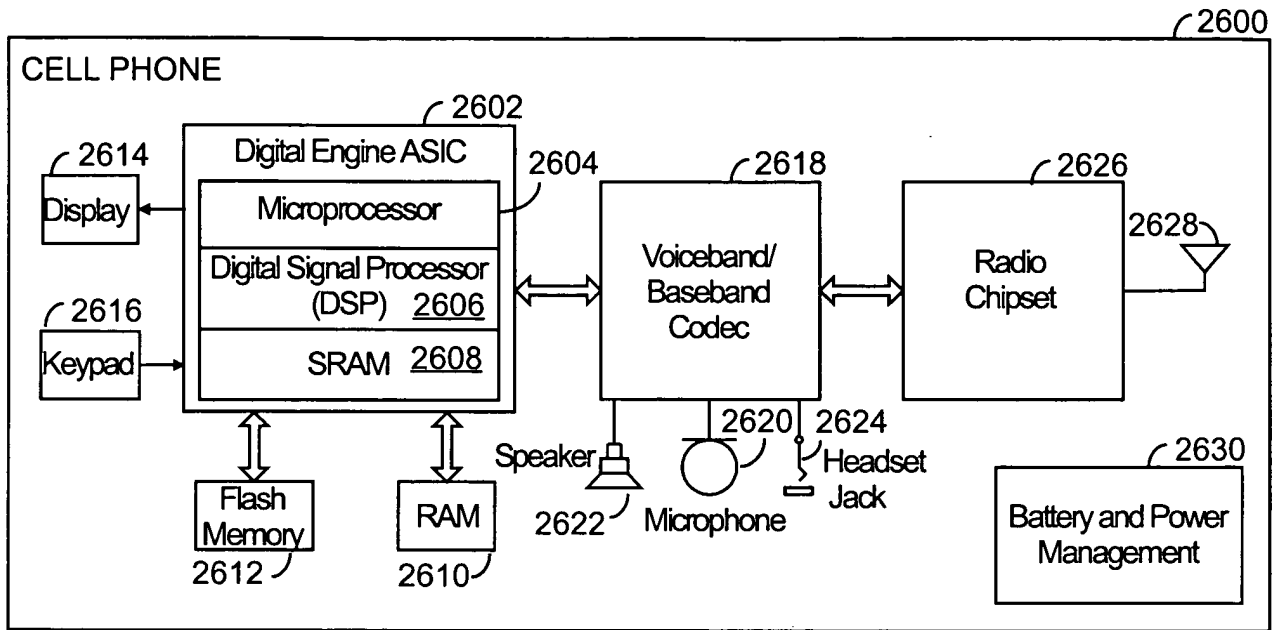


FIG. 26

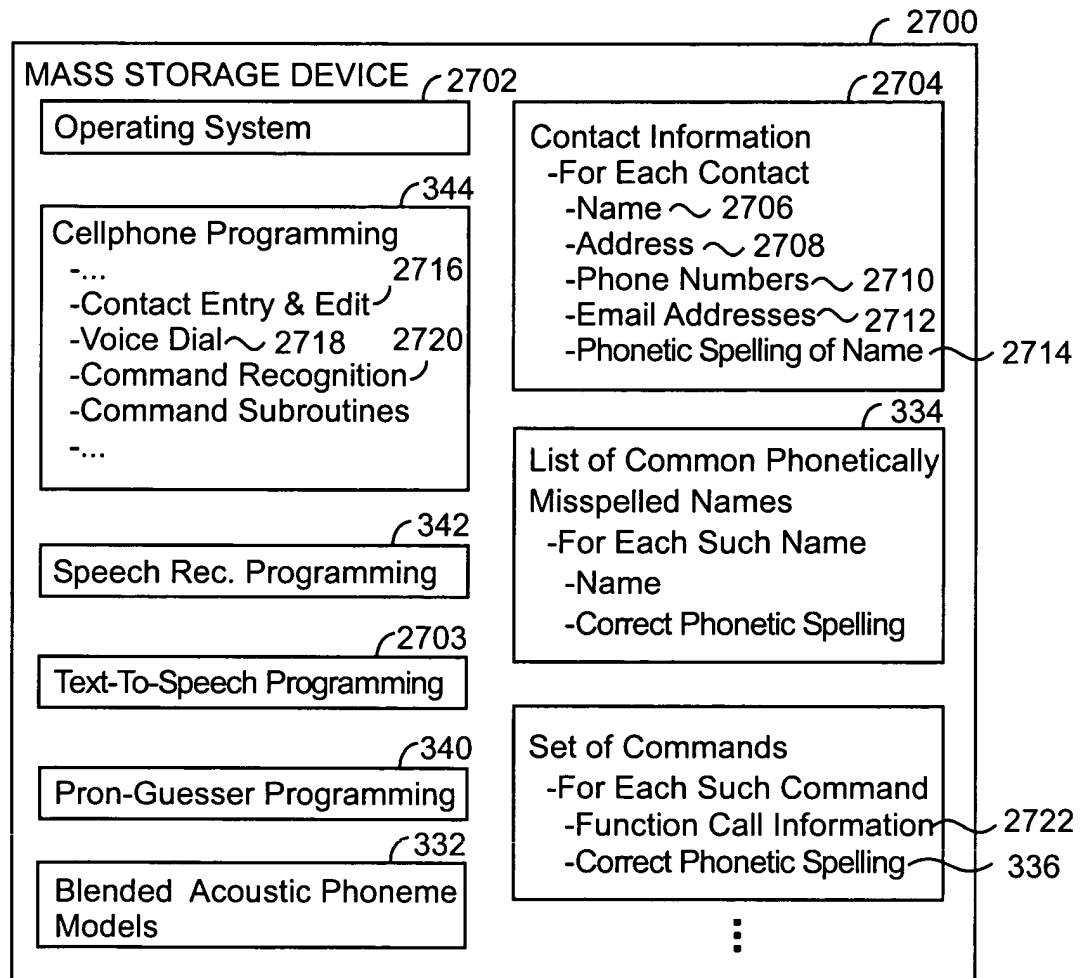


FIG. 27

-cell phone control programming~334

-...

-if user selects phone contact entry mode~2802

-call name entry routine to enable user to enter name~2804

-if user enters a spelling of a name~2806

-create a blank phone contact record~2808

-enter spelling of name in name field of record~2810

-if spelling is one for which a correct phonetic spelling has been stored~2812

-place correct phonetic spelling in phonetic spelling field of record~2814

-else~2815

-call pron-guesser for name~2816

-place phonetic spelling generated by pron-guesser in phonetic spelling field for record~2818

-call phone number entry routine to enable user to enter phone number~2820

-...

-if user enters a phone number~2822

-place entered phone number in phone number field of record~2824

-...

-...

-...

-if user selects voice dial~2826

-call the speech recognition programming with a vocabulary corresponding the names and corresponding phonetic spellings, whether guess or known correct, in the contact records and one or more command words and their known correct spellings~2828

-if best match is a name in a contact record~2830

-if there is a phone number stored in the contact record~2832

-use text-to-speech synthesizer to generate sound of recognized name using its phonetic spelling~2833

-dial that number~2834

-else if best match is a command~2836

-call subroutine associated with the given command~2838

-...

-if in command recognition mode~2840

-call the speech recognition programming with a vocabulary corresponding to the commands recognizable in the command mode~2842

-if best match is a given command~2844

-call subroutine associated with the given command~2846

2716

2718

2720

FIG. 28

- speech recognition programming~342
 - if detect an utterance~2902
 - score the match between sounds of the utterance and the sequence of blended acoustic phoneme models corresponding with the phonetic spelling of each word in the vocabulary for which the recognition program has been called~2906
 - return with an indication of one or more words having the best score in the match~2908

FIG. 29

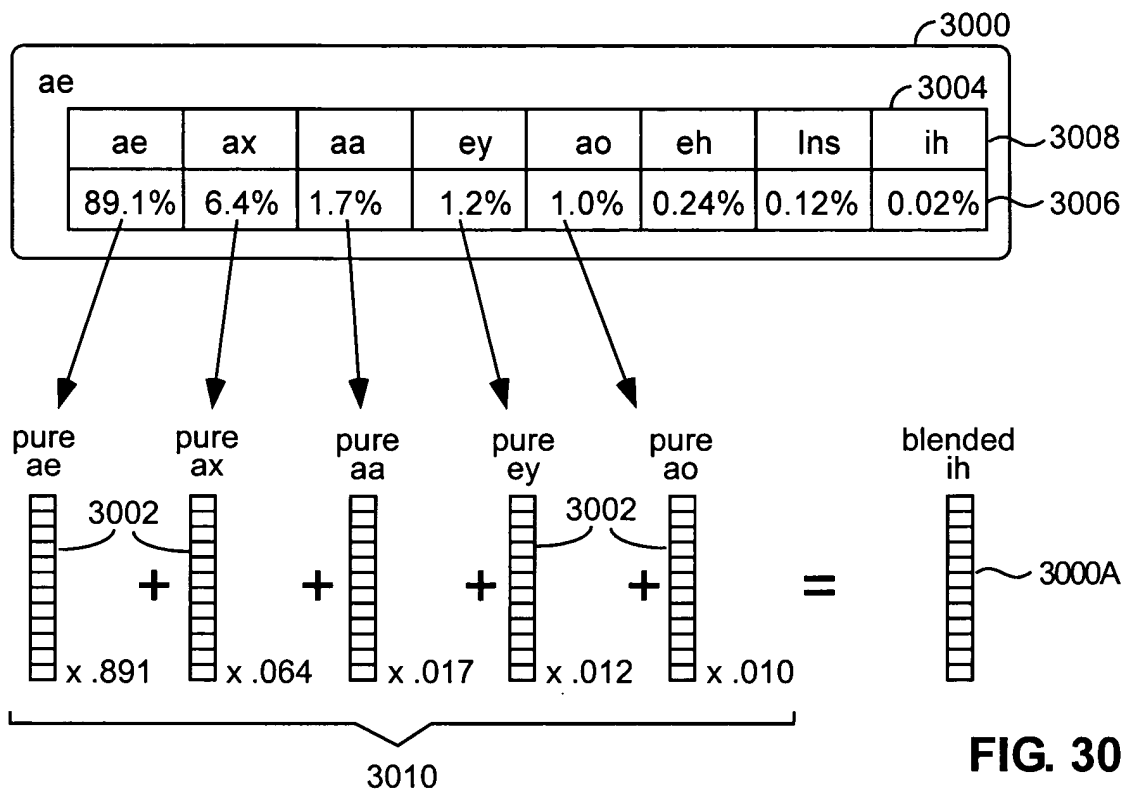


FIG. 30

- training pure acoustic phoneme models~1600A
 - for each of a plurality of acoustic training words~1602
 - OBTAIN A CORRECT PHONETIC SPELLING FOR THE WORD~1604A
 - create an untrained associate word model for the word, composed of a sequence of untrained acoustic phoneme models corresponding to the word's phonetic spelling~1606
 - repeatedly~1608
 - use HMM to map one or more utterances of each training word against the acoustic model for that word~1610
 - alter each acoustic phoneme model to be more like the utterance sounds mapped against it in all training words in which it occurs~1612

FIG. 31

- training segregated blended models~3200
 - for each phoneme~3202
 - create an empty mapping list for the phoneme~3204
 - for each of a plurality of acoustic training words~3206
 - have pron-guesser guess phonetic spelling for the word~3208
 - map phonemes of guessed phonetic spelling for word against correct phonetic spelling for word~3210
 - for each guessed phoneme in guessed phonetic spelling~3212
 - find the sequence of one or more correct phonemes mapped against the guessed phoneme~3214
 - if the sequence mapped against the guessed phoneme does not already have an entry in the guessed phoneme's mapping list~3216
 - create an entry in that mapping list for the sequence of one or more correct phonemes and set its count to one~3218
 - else~3220
 - increment the count of the entry in the guessed phoneme's mapping list corresponding to the sequence of one or more correct phonemes~3222
 - for each phoneme~3224
 - normalize the count of its mapping list, so that the total of the count associated with all phonemes in the list equals one~3226

FIG. 32

- adapting pure acoustic models~3300
 - prompt user to say utterances of one or more adaptation training words~3302
 - map utterances of each adaptation training word against a sequence of pure acoustic phoneme models corresponding to a correct phonetic spelling of that word~3304
 - use the sounds mapped against each pure acoustic phoneme models to adapt that model~3306
 - for each given phoneme~3308
 - create an empty blended acoustic phoneme model for the given phoneme~3310
 - for each phoneme listed in the given phoneme's mapping list having a count above a certain value~3312
 - multiply the pure model of the listed phoneme by the listed phoneme's count in the mapping list~3314
 - add the resulting product to blended acoustic phoneme model for the given phoneme~3316

FIG. 33

- speech recognizing against segregated acoustic models~342A
 - if detect an utterance~2902
 - score the match between sounds of the utterance and the sequence of SEGREGATED blended acoustic phoneme models corresponding with the phonetic spelling of each word of the vocabulary for which the recognition program has been called~2906A
 - return with an indication of one or more words having the best score in the match~2908

FIG. 34

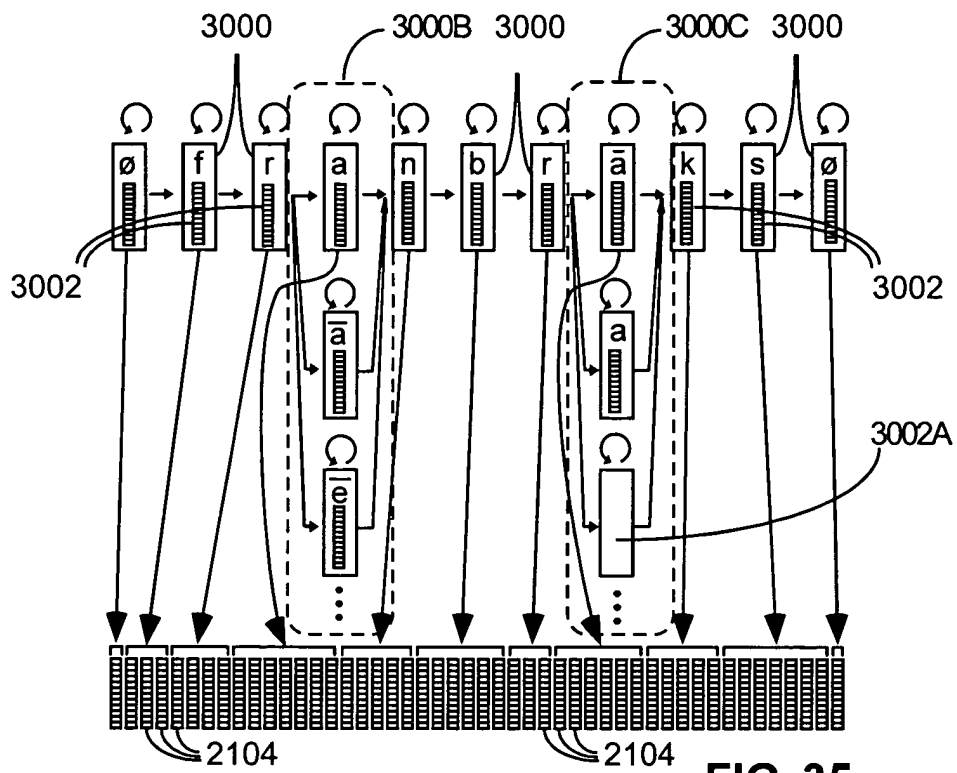


FIG. 35

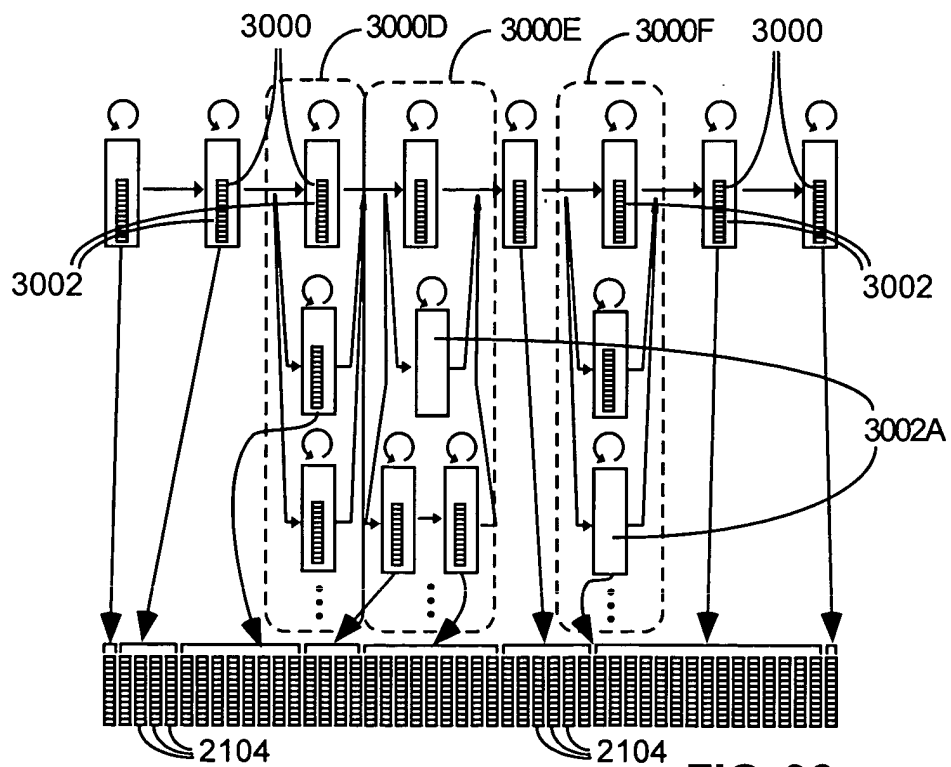


FIG. 36